

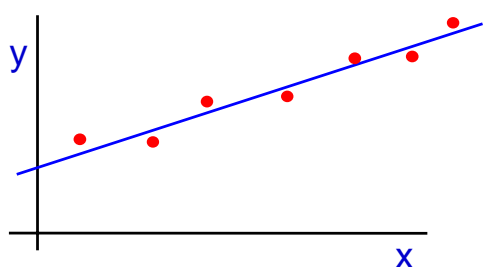
Lectures 3 & 4: Estimators

Hilary Term 2007

A. Zisserman

- Maximum likelihood estimation (MLE)
 - Likelihood function, information matrix
- Least Squares estimation
 - Total least squares
 - General least squares
 - Regularization
- Maximum a Posteriori estimation (MAP)
 - Posterior density via Bayes' rule
 - Confidence regions

Maximum Likelihood Estimation



$$\min_{a,b} \sum_i^n (y_i - (ax_i + b))^2$$

In the line fitting (linear regression) example the estimate of the line parameters θ involved two steps:

1. Write down the likelihood function expressing the probability of the data \mathbf{z} given the parameters θ
2. Maximize the likelihood to determine θ

i.e. choose the value of θ so as to make the data as likely as possible.

Note, the solution does not depend on the variance σ^2 of the noise

Example application: Navigation

Take multiple measurements of beacons to improve location estimates

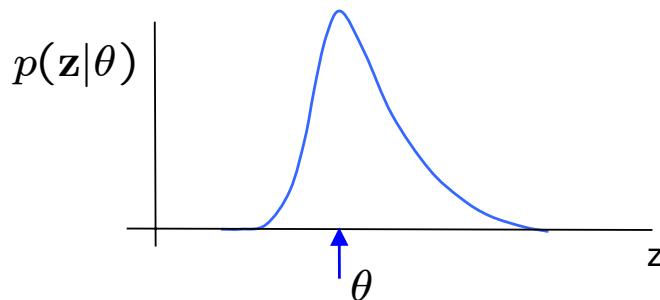


The Likelihood function

More formally, the likelihood is defined as

$$\mathcal{L}(\theta) = p(\mathbf{z}|\theta)$$

- the likelihood is the conditional probability of obtaining measurements $\mathbf{z} = (z_1, z_2, \dots, z_n)$ given that the true value is θ



Note, in general

- The likelihood is a function of θ , but it is **not** a probability distribution over θ , and its integral with respect to θ does not (necessarily) equal one.
- It would be incorrect to refer to this as “the likelihood of the data”.

Given observations \mathbf{z} and a likelihood function $\mathcal{L}(\theta)$, the **Maximum Likelihood** estimate of θ is the value of θ which maximizes the likelihood function

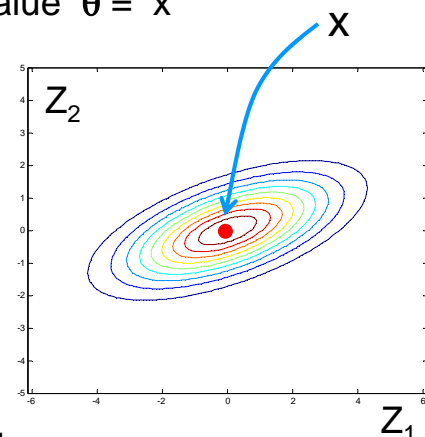
$$\hat{\theta} = \arg \max_{\theta} \mathcal{L}(\theta)$$

Example: Gaussian Sensors

Likelihood is a Gaussian response in 2D for true value $\theta = \mathbf{x}$

$$p(\mathbf{z}|\mathbf{x}) = \frac{1}{C} \exp \left\{ -\frac{1}{2}(\mathbf{z} - \mathbf{x})^\top \Sigma^{-1}(\mathbf{z} - \mathbf{x}) \right\}$$

$$\mathcal{L}(\theta) = p(\mathbf{z}|\theta)$$



Now, look at several problems involving combining multiple measurements from Gaussian sensors

MLE Example 1

Suppose we have two independent sonar measurements (z_1, z_2) of position x , and the sensor error may be modelled as $p(z_i|\theta) = N(\theta, \sigma^2)$, determine the MLE of x .

Solution

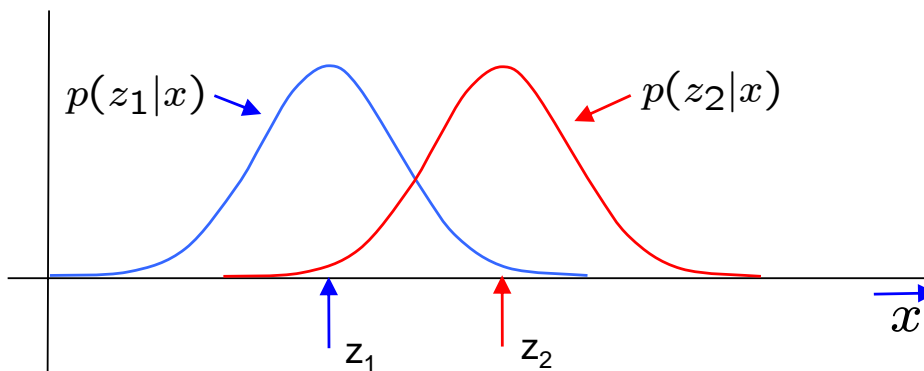
Since the sensors are independent the likelihood is

$$\mathcal{L}(x) = p(z_1, z_2|x) = p(z_1|x) p(z_2|x)$$

and since the sensors are Gaussian

$$\mathcal{L}(x) \sim e^{-\frac{(z_1-x)^2}{2\sigma^2}} \times e^{-\frac{(z_2-x)^2}{2\sigma^2}} = e^{-\frac{(z_1-x)^2 + (z_2-x)^2}{2\sigma^2}}$$

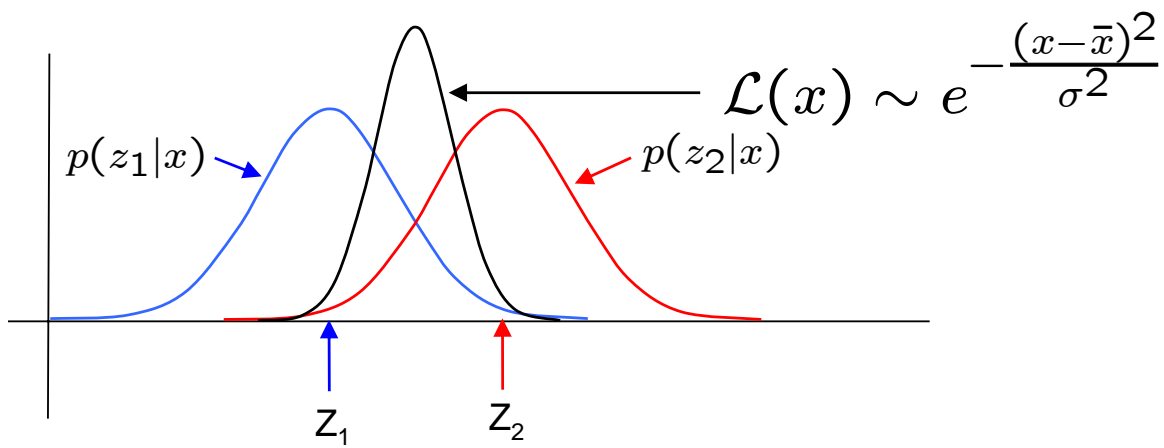
ignore irrelevant normalization constants



$$\mathcal{L}(x) \sim e^{-\frac{(z_1-x)^2 + (z_2-x)^2}{2\sigma^2}}$$

$$\begin{aligned} -\ln \mathcal{L}(x) &= ((z_1 - x)^2 + (z_2 - x)^2)/(2\sigma^2) \\ &= (2x^2 - 2x(z_1 + z_2) + z_1^2 + z_2^2)/(2\sigma^2) \\ &= (x - \bar{x})^2/(\sigma^2) + c(z_1, z_2) \end{aligned}$$

$$\mathcal{L}(x) \sim e^{-\frac{(x-\bar{x})^2}{\sigma^2}} \quad \text{with} \quad \bar{x} = \frac{z_1 + z_2}{2}$$



Note

- the likelihood is a Gaussian
- the variance is reduced (cf the original sensor variance)

Maximum likelihood estimate of x

$$\hat{x} = \arg \max_x \mathcal{L}(x)$$

negative log likelihood

$$-\ln \mathcal{L}(x) = (x - \bar{x})^2 / (\sigma^2) + c(z_1, z_2)$$

$$\text{ML } \hat{x} = \arg \min_x \{-\ln \mathcal{L}(x)\}$$

Compute min by differentiating wrt x

$$\frac{d\{-\ln \mathcal{L}(x)\}}{dx} = 2(x - \bar{x}) / (\sigma^2) = 0$$

$$\hat{x} = \bar{x} = \frac{z_1 + z_2}{2}$$

MLE Example 2: Gaussian fusion with different variances

Suppose we have two independent sonar measurements (z_1, z_2) of position x , and the sensor error may be modelled as $p(z_1|\theta) = N(\theta, \sigma_1^2)$ and $p(z_2|\theta) = N(\theta, \sigma_2^2)$, determine the MLE of x .

Solution

Again, as the sensors are independent

$$\mathcal{L}(x) = p(z_1, z_2|x) = p(z_1|x) p(z_2|x)$$

$$\mathcal{L}(x) \sim e^{-\frac{(z_1-x)^2}{2\sigma_1^2}} \times e^{-\frac{(z_2-x)^2}{2\sigma_2^2}}$$

The negative log likelihood is then:

$$\begin{aligned} -\ln \mathcal{L}(x) &= \frac{1}{2} [\sigma_1^{-2}(z_1 - x)^2 + \sigma_2^{-2}(z_2 - x)^2] + \text{const} \\ &= \frac{1}{2} [(\sigma_1^{-2} + \sigma_2^{-2})x^2 - 2(\sigma_1^{-2}z_1 + \sigma_2^{-2}z_2)x] + \text{const} \\ &= \frac{1}{2}(\sigma_1^{-2} + \sigma_2^{-2}) \left[x - \frac{\sigma_1^{-2}z_1 + \sigma_2^{-2}z_2}{\sigma_1^{-2} + \sigma_2^{-2}} \right]^2 + \text{const} \end{aligned}$$

which is maximised w.r.t. x when

$$x_{\text{MLE}} = \frac{\sigma_1^{-2}z_1 + \sigma_2^{-2}z_2}{\sigma_1^{-2} + \sigma_2^{-2}}$$

e.g. if the sensors are: $p(z_1|x) \sim N(x, 10^2)$ $p(z_2|x) \sim N(x, 20^2)$

Suppose we obtain sensor readings of $z_1 = 130, z_2 = 170$.

$$\hat{x}_{\text{MLE}} = \frac{130/10^2 + 170/20^2}{1/10^2 + 1/20^2} = 138.0$$

- the ML estimate is closer to the more confident measurement

The Information Matrix

We have seen that for two independent Normal variables:

- information weighted mean

$$\hat{x} = \frac{\sigma_1^{-2} z_1 + \sigma_2^{-2} z_2}{\sigma_1^{-2} + \sigma_2^{-2}}$$

- additivity of statistical information

$$\sigma_x^{-2} = \sigma_1^{-2} + \sigma_2^{-2}$$

(like the parallel resistors formula)

This motivates the definition of the **information matrix** $S = \Sigma^{-1}$

$$S_x = S_{z_1} + S_{z_2}$$

In 2D with $\mathbf{z} = (z_x, z_y)^\top$

$$S_x = \Sigma_x^{-1} = S_{\mathbf{z}_1} + S_{\mathbf{z}_2}$$

$$\hat{\mathbf{x}} = \Sigma_x (S_{\mathbf{z}_1} \mathbf{z}_1 + S_{\mathbf{z}_2} \mathbf{z}_2)$$

$$\text{If } \Sigma = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad \text{then } S = \begin{bmatrix} 1/\sigma_x^2 & 0 \\ 0 & 1/\sigma_y^2 \end{bmatrix}$$

- large covariance \rightarrow uncertainty, *lack of information*
- small covariance \rightarrow some confidence, or *strong information*

A singular covariance matrix implies “infinite” or “perfect” information.

MLE Example 3: Gaussian fusion with 2D sensors

Suppose we have two independent 2D vision measurements ($\mathbf{z}_1, \mathbf{z}_2$) of position \mathbf{x} , and the sensor error may be modelled as $p(\mathbf{z}_1|\theta) = \mathcal{N}(\theta, \Sigma_1)$ and $p(\mathbf{z}_2|\theta) = \mathcal{N}(\theta, \Sigma_2)$, determine the MLE of \mathbf{x} , where

$$\mathbf{z}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} \quad \text{and} \quad \mathbf{z}_2 = \begin{pmatrix} 2 \\ -1 \end{pmatrix} \quad \Sigma_2 = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$$

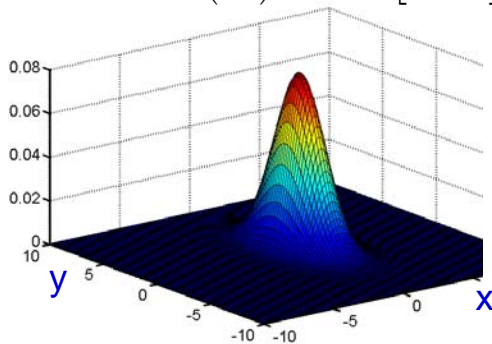
Solution

Again, the sensors are independent so we **sum** their information

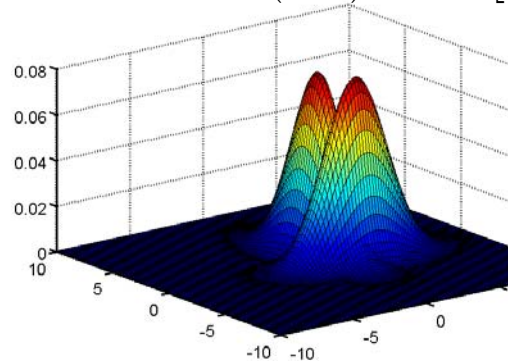
$$\Sigma_x^{-1} = \Sigma_1^{-1} + \Sigma_2^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 0.25 \end{bmatrix} + \begin{bmatrix} 0.25 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1.25 & 0 \\ 0 & 1.25 \end{bmatrix}$$

$$\begin{aligned} \hat{\mathbf{x}} &= \Sigma_x (S_{\mathbf{z}_1} \mathbf{z}_1 + S_{\mathbf{z}_2} \mathbf{z}_2) \\ &= \begin{bmatrix} 0.8 & 0 \\ 0 & 0.8 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 \\ 0 & 0.25 \end{bmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{bmatrix} 0.25 & 0 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} 2 \\ -1 \end{pmatrix} \right) = \begin{pmatrix} 1.2 \\ -0.6 \end{pmatrix} \end{aligned}$$

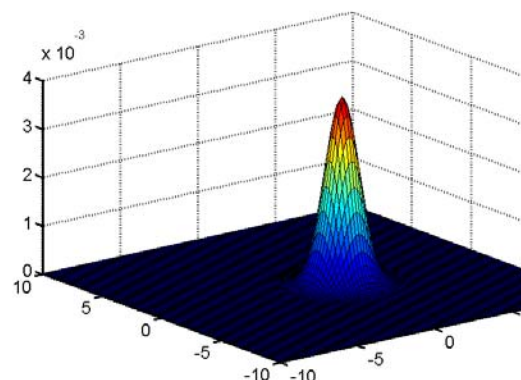
$$\mathcal{N}(\mathbf{z}_1, \Sigma_1), \quad \mathbf{z}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \Sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}$$



$$\text{and } \mathcal{N}(\mathbf{z}_2, \Sigma_2), \quad \mathbf{z}_2 = \begin{pmatrix} 2 \\ -1 \end{pmatrix} \quad \Sigma_2 = \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix}$$



$$\mathcal{L}(\mathbf{x}) = p(\mathbf{z}_1|\mathbf{x}) p(\mathbf{z}_2|\mathbf{x})$$



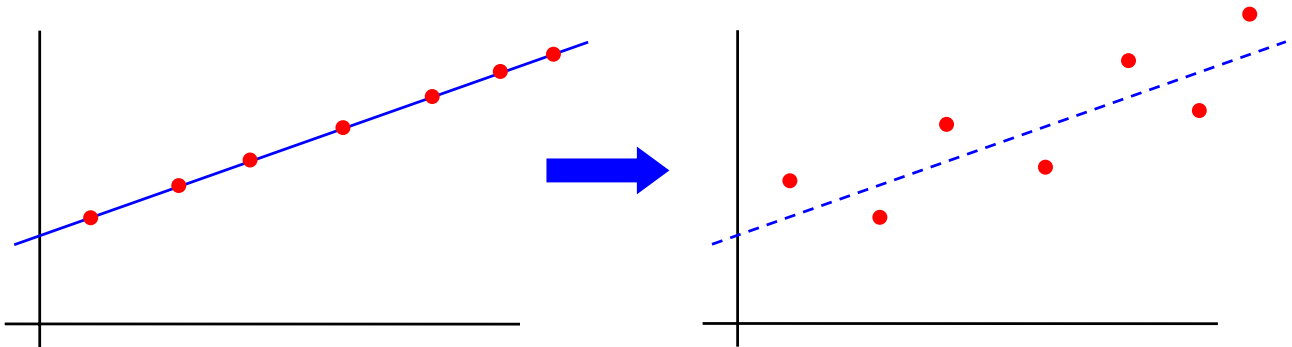
Line fitting with errors in both coordinates

Suppose points are measured from a true line

$\tilde{y} = a\tilde{x} + b$ with errors as

$$x_i = \tilde{x}_i + v_i \quad v_i \sim N(0, \sigma^2)$$

$$y_i = \tilde{y}_i + w_i \quad w_i \sim N(0, \sigma^2)$$



In the case of linear regression, the ML estimate gave the cost function

$$\min_{a,b} \sum_i^n (y_i - (ax_i + b))^2$$

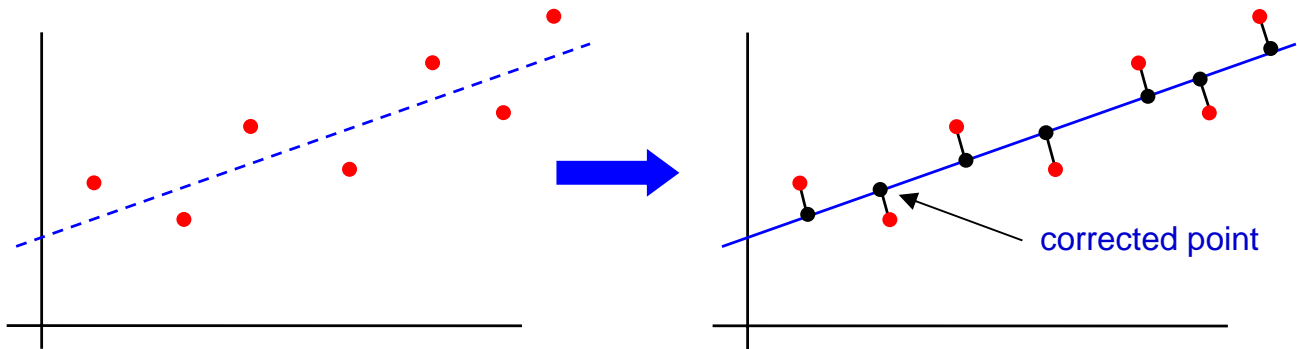
Diagram illustrating the cost function components:

- y_i is labeled "measured value" (indicated by a blue arrow pointing to y_i).
- $(ax_i + b)$ is labeled "predicted value" (indicated by a blue bracket and arrow pointing to the expression).

In the case of errors in both coordinates, the ML cost function is

$$\min_{a,b} \sum_i^n (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \quad \text{subject to} \quad \hat{y}_i = a\hat{x}_i + b$$

i.e. it is necessary also to estimate (predict)
corrected points (\hat{x}_i, \hat{y}_i)



$$\min_{a,b} \sum_i^n (x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \quad \text{subject to} \quad \hat{y}_i = a\hat{x}_i + b$$

This estimator is called **total least squares**, or **orthogonal regression**

Solution **[exercise – non-examinable]**

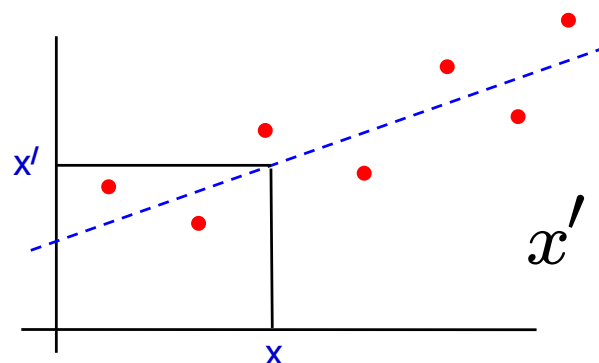
- line passes through centroid of the points $\mu = \sum_i^n \mathbf{x}_i$
- line direction is parallel to the eigenvector of the covariance matrix corresponding to the least eigenvalue

$$\Sigma = \sum_i^n [(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^\top]$$

Application: Estimating Geometric Transformations

The line estimation problem is equivalent to estimating a 1D affine transformation from noisy point correspondences

$$\{x'_i \leftrightarrow x_i\}$$



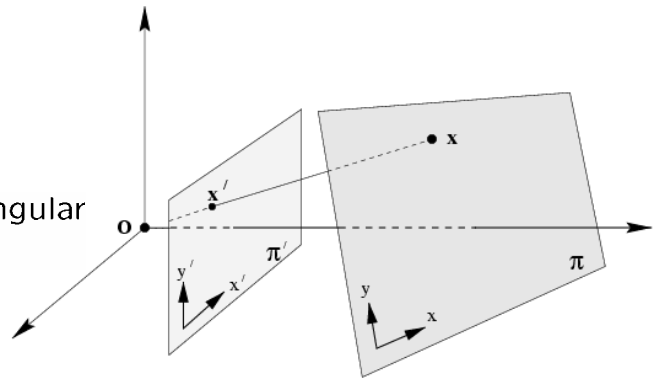
$$x' = \alpha x + \beta$$

Can also compute the ML estimate of 2D transformations

Reminder: Plane Projective transformation

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

or $\mathbf{x}' = \mathbf{H}\mathbf{x}$, where \mathbf{H} is a 3×3 non-singular homogeneous matrix.

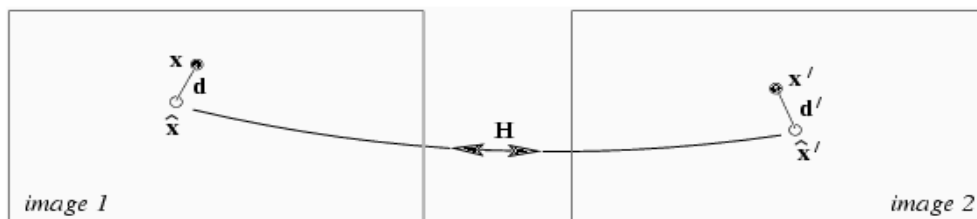


- A projective transformation is also called a "homography" and a "collineation".
- \mathbf{H} has 8 degrees of freedom.
- It can be estimated from 4 or more point correspondences

Maximum Likelihood estimation of homography \mathbf{H}

If the measurement error is Gaussian, then the ML estimate of \mathbf{H} **and** the corrected correspondences $\{\hat{\mathbf{x}}_i \leftrightarrow \hat{\mathbf{x}}'_i\}$ is given by minimizing

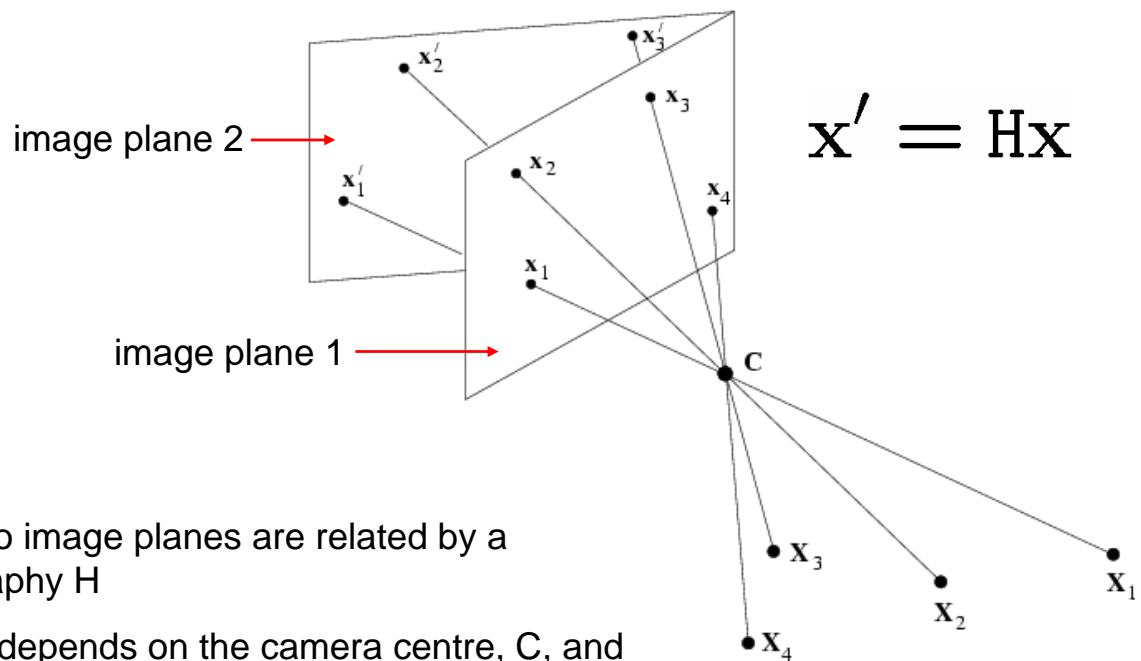
$$\mathcal{C} = \sum_i d^2(\hat{\mathbf{x}}_i, \mathbf{x}_i) + d^2(\hat{\mathbf{x}}'_i, \mathbf{x}'_i) \quad \text{subject to} \quad \hat{\mathbf{x}}_i = \hat{\mathbf{H}}\hat{\mathbf{x}}'_i, \forall i$$



Cost function minimization:

- for 2D affine transformations there is a matrix solution (**non-examinable**)
- for 2D projective transformation, minimize numerically using e.g. non-linear gradient descent

e.g. Camera rotating about its centre



- The two image planes are related by a homography H
- H only depends on the camera centre, C , and the planes, **not** on the 3D structure

Example : Building panoramic mosaics

4 frames from a sequence of 30



The camera rotates (and zooms) with fixed centre

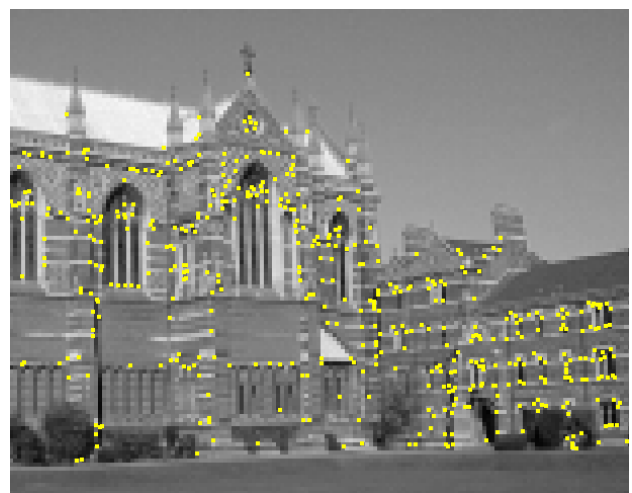
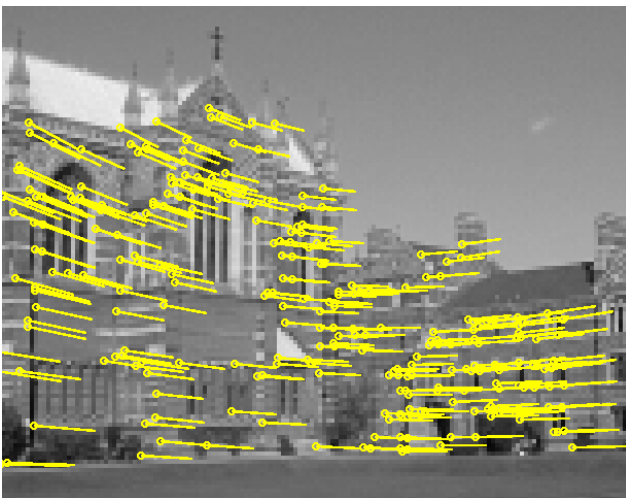
Example : Building panoramic mosaics

30 frames



The camera rotates (and zooms) with fixed centre

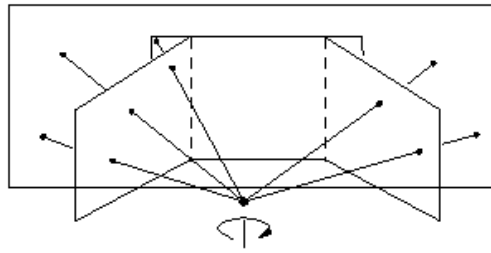
Homography between two frames



ML estimate of homography from these 100s of point correspondences

Choice of mosaic frame

Choose central image as reference



This produces the classic "bow-tie" mosaic.



General Linear Least Squares

$$\min_{a,b} \sum_i (y_i - ax_i - b)^2$$

Write the residuals as an n -vector

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} - \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_n & 1 \end{bmatrix} \begin{pmatrix} a \\ b \end{pmatrix}$$

$$\mathbf{w} = \mathbf{y} - H\theta$$

H is a $n \times 2$ matrix

Then the sum of squared residuals becomes

$$\mathbf{w}^\top \mathbf{w} = (\mathbf{y} - H\theta)^\top (\mathbf{y} - H\theta)$$

We want to minimize this w.r.t. θ

$$\frac{d}{d\theta} (\mathbf{y} - H\theta)^\top (\mathbf{y} - H\theta) = 2H^\top (\mathbf{y} - H\theta) = 0$$

$$H^\top H\theta = H^\top \mathbf{y}$$

$$\theta = (H^\top H)^{-1} H^\top \mathbf{y} = H^+ \mathbf{y}$$

$$\begin{pmatrix} \end{pmatrix} = \begin{pmatrix} \end{pmatrix} \begin{pmatrix} \end{pmatrix} \begin{pmatrix} \end{pmatrix}$$

Summary

- If the generative model is linear $\mathbf{y} = H\theta$
- then the ML solution for Gaussian noise is $\theta = H^+ \mathbf{y}$
- where the matrix $H^+ = (H^\top H)^{-1} H^\top$ is the **pseudo-inverse** of H

Generalization I: multi-linear regression

$$y_i = \theta_2 x_i + \theta_1 z_i + \theta_0 + w_i$$

$$H = \begin{bmatrix} x_1 & z_1 & 1 \\ x_2 & z_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & z_n & 1 \end{bmatrix} \quad \theta = (H^T H)^{-1} H^T y$$

More general still with basis functions $\phi_j(x)$

$$y_i = \sum_j \theta_j \phi_j(x_i) + w_i$$

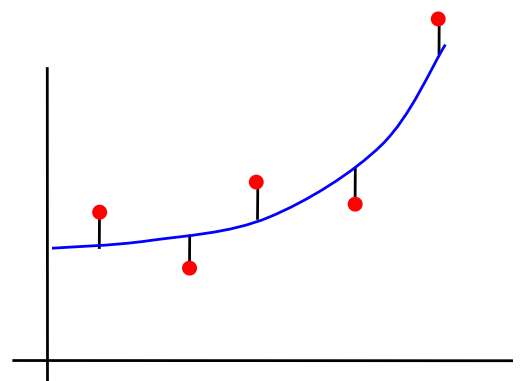
Generalization II: regression with basis functions

$$y_i = \sum_j \theta_j \phi_j(x_i) + w_i$$

for example, quadratic regression

$$y_i = \theta_2 (x_i)^2 + \theta_1 (x_i) + \theta_0 + w_i$$

$$H = \begin{bmatrix} \phi_2(x_1) & \phi_1(x_1) & 1 \\ \phi_2(x_2) & \phi_1(x_2) & 1 \\ \vdots & \vdots & \vdots \\ \phi_2(x_n) & \phi_1(x_n) & 1 \end{bmatrix} = \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n^2 & x_n & 1 \end{bmatrix}$$



- what is the correct function to fit ?
- how is the quality of the estimate assessed?

Summary point

We have seen:

- An estimator is a function $f(z)$ which computes an estimate

$$\hat{\theta} = f(\mathbf{z})$$

for the parameters θ from measurements $\mathbf{z} = (z_1, z_2, \dots, z_n)$

- The Maximum Likelihood estimator is

$$\hat{\theta} = \arg \max_{\theta} p(\mathbf{z}|\theta)$$

Now, we will compute the posterior, and hence the [Maximum a Posteriori \(MAP\)](#) estimator

Bayes' rule and the Posterior distribution

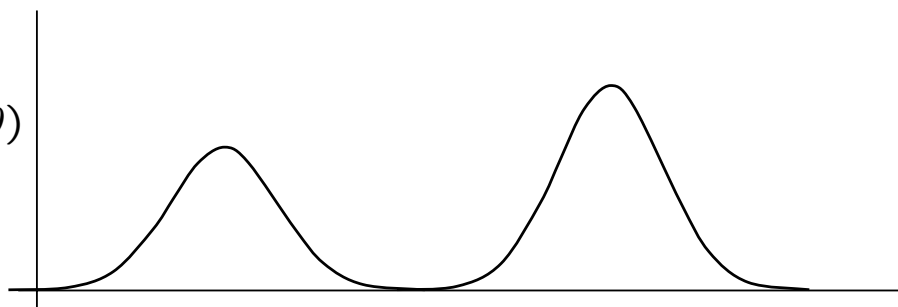
Suppose we have

- $p(z|\theta)$ the conditional density of the sensor(s)
- additionally $p(\theta)$, a prior density of θ the parameter to be estimated.

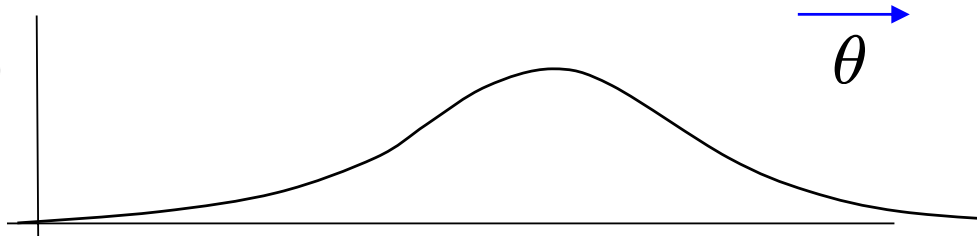
Use Bayes' rule to get the **posterior** density

$$p(\theta|z) = \frac{p(z|\theta)p(\theta)}{p(z)} = \frac{\text{likelihood} \times \text{prior}}{\text{normalizing factor}}$$

Likelihood $p(z|\theta)$

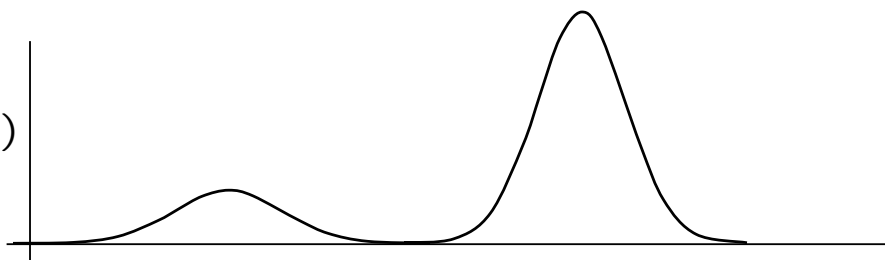


Prior $p(\theta)$

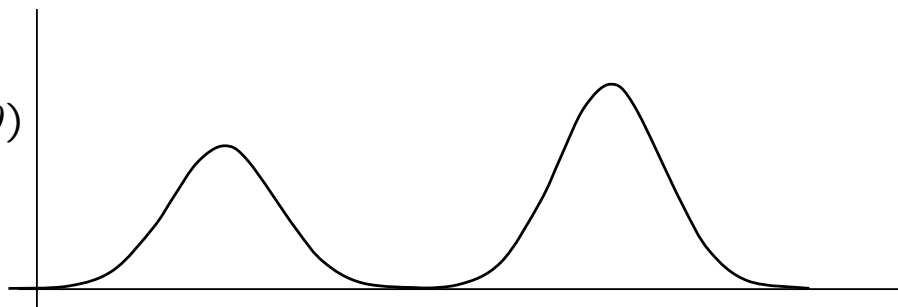


Posterior

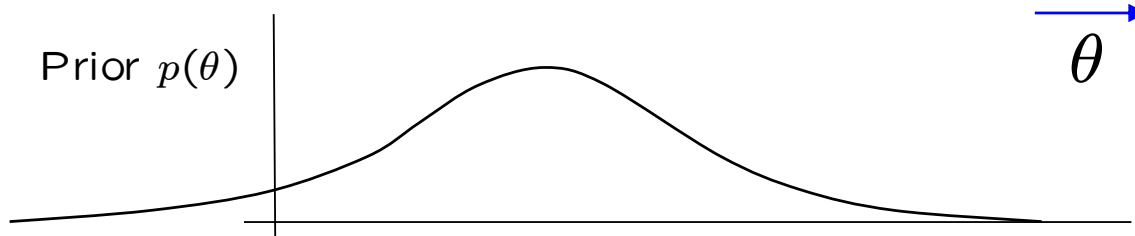
$p(\theta|z) \sim p(z|\theta)p(\theta)$



Likelihood $p(z|\theta)$

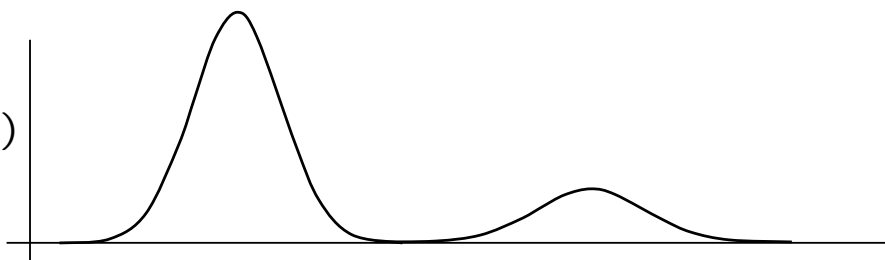


Prior $p(\theta)$

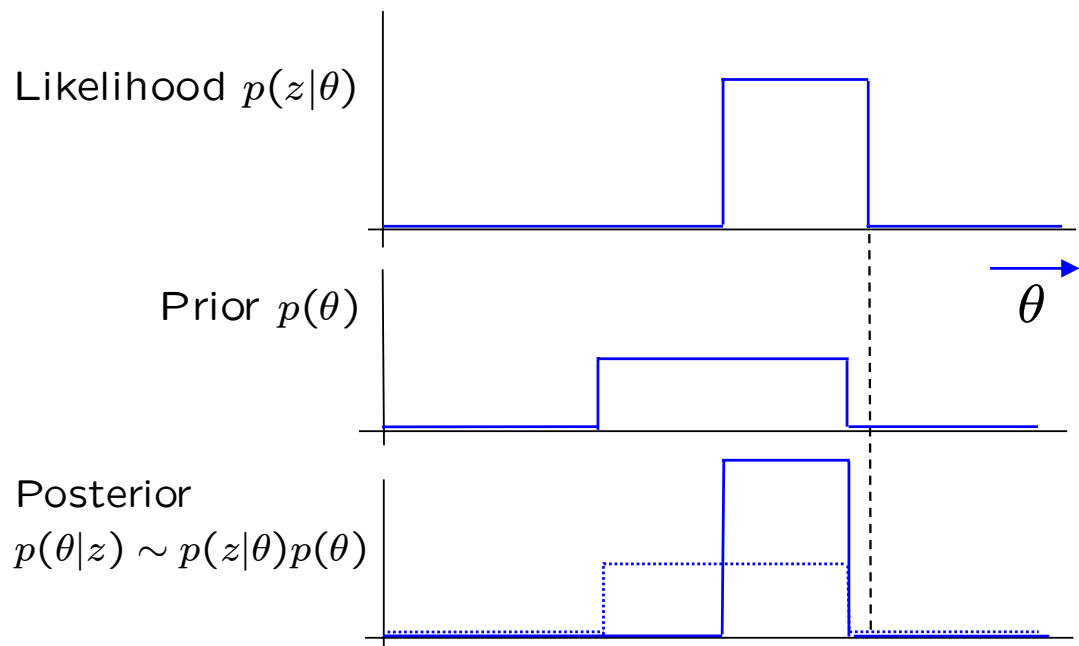


Posterior

$p(\theta|z) \sim p(z|\theta)p(\theta)$

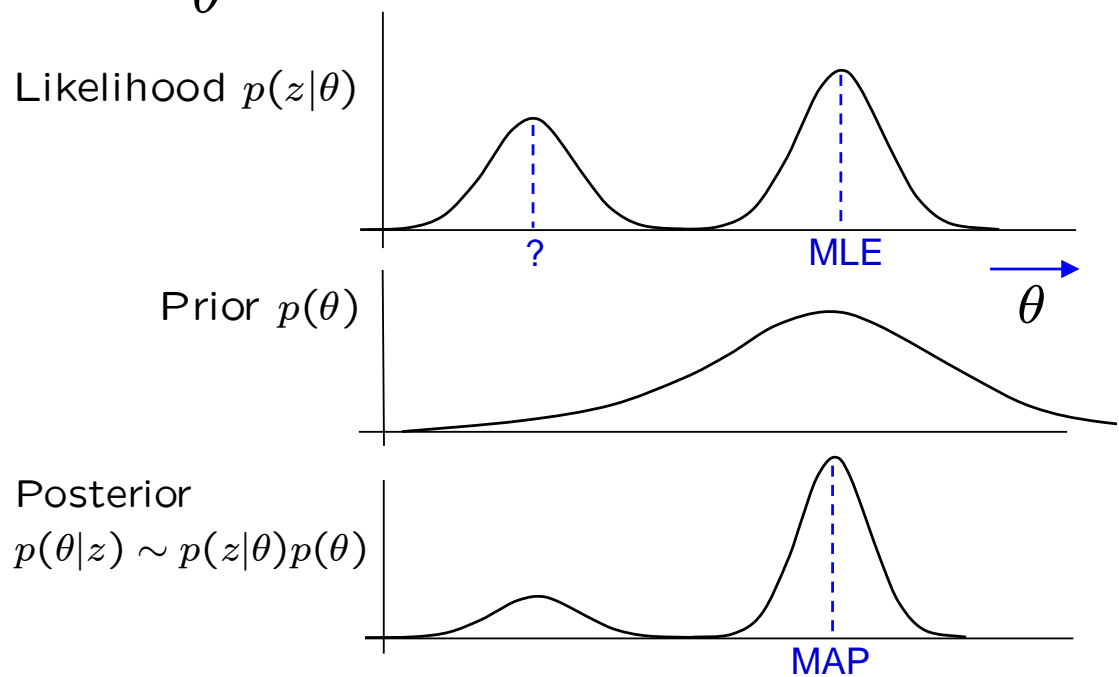


Example: posterior for uniform distributions



MAP Estimator

$$\hat{\theta} = \arg \max_{\theta} p(\theta|z)$$



MAP Example: Gaussian fusion

Consider again two sensors modelled as independent normal distributions:

$$\begin{aligned}z_1 &= \theta + w_1, \quad p(z_1|\theta) \sim N(\theta, \sigma_1^2) \\z_2 &= \theta + w_2, \quad p(z_2|\theta) \sim N(\theta, \sigma_2^2)\end{aligned}$$

so that the joint likelihood is

$$p(z_1, z_2|\theta) = p(z_1|\theta) \times p(z_2|\theta)$$

Suppose in addition we have prior information that

$$p(\theta) \sim N(\theta_p, \sigma_p^2)$$

Then the posterior density is

$$p(\theta|z_1, z_2) \sim p(z_1, z_2|\theta) \times p(\theta)$$

The posterior is a Gaussian, and we can write down its mean and variance simply by adding statistical information:

$$p(\theta|z_1, z_2) \sim N(\mu_{\text{pos}}, \sigma_{\text{pos}}^2)$$

where

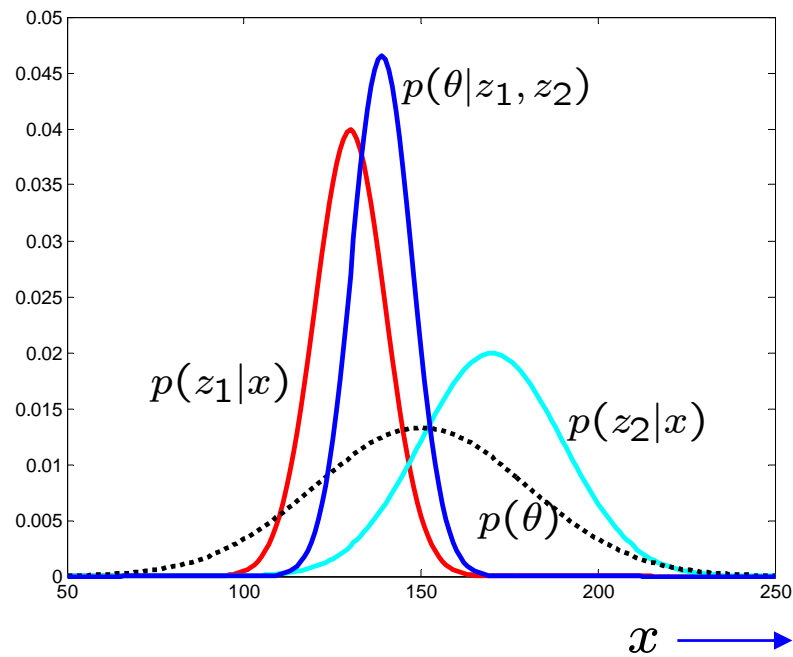
$$\begin{aligned}\sigma_{\text{pos}}^{-2} &= \sigma_1^{-2} + \sigma_2^{-2} + \sigma_p^{-2} \\ \mu_{\text{pos}} &= \frac{\sigma_1^{-2}z_1 + \sigma_2^{-2}z_2 + \sigma_p^{-2}\theta_p}{\sigma_1^{-2} + \sigma_2^{-2} + \sigma_p^{-2}} = \hat{\theta}_{\text{MAP}}\end{aligned}$$

e.g. if the sensors are: $p(z_1|x) \sim N(x, 10^2)$, $p(z_2|x) \sim N(x, 20^2)$, and the prior is $p(x) \sim N(150, 30^2)$.

Suppose we obtain sensor readings of $z_1 = 130, z_2 = 170$ then

$$\begin{aligned}\hat{x}_{\text{MAP}} &= \frac{130/10^2 + 170/20^2 + 150/30^2}{1/10^2 + 1/20^2 + 1/30^2} = 139.0 \\ \hat{\sigma}_{\text{pos}} &= \frac{1}{\sqrt{1/10^2 + 1/20^2 + 1/30^2}} = 8.57\end{aligned}$$

Matlab example



```
inc=1; t = [50:inc:250];  
pz1 = normal(t, 130,10^2); pz2 = normal(t, 170,20^2); prx = normal(t, 150,30^2);  
posx = pz1 .* pz2 .* prx / (inc*(sum(pz1 .* pz2 .* prx)));  
plot(t,pz1,'r', t,pz2,'g', t,prx,'k:', t, posx, 'b');
```

MAP vs MLE

For the posterior

$$\sigma_{\text{pos}}^{-2} = \sigma_1^{-2} + \sigma_2^{-2} + \sigma_p^{-2} \quad \hat{\theta}_{\text{MAP}} = \frac{\sigma_1^{-2} z_1 + \sigma_2^{-2} z_2 + \sigma_p^{-2} \theta_p}{\sigma_1^{-2} + \sigma_2^{-2} + \sigma_p^{-2}}$$

Recall that for the likelihood

$$\sigma_L^{-2} = \sigma_1^{-2} + \sigma_2^{-2} \quad \hat{\theta}_{\text{MLE}} = \frac{\sigma_1^{-2} z_1 + \sigma_2^{-2} z_2}{\sigma_1^{-2} + \sigma_2^{-2}}$$

so

$$\sigma_{\text{pos}}^{-2} = \sigma_L^{-2} + \sigma_p^{-2} \quad \hat{\theta}_{\text{MAP}} = \frac{\sigma_p^{-2} \theta_p + \sigma_L^{-2} \hat{\theta}_{\text{MLE}}}{\sigma_p^{-2} + \sigma_L^{-2}}$$

i.e. prior acts as an additional sensor

- Decrease in posterior variance relative to the prior and likelihood

$$\frac{1}{\sigma_{\text{pos}}^2} = \frac{1}{\sigma_L^2} + \frac{1}{\sigma_p^2}$$

e.g.

$$\sigma_L = \frac{1}{\sqrt{1/10^2 + 1/20^2}} = 8.94 \quad \sigma_p = 30 \quad \sigma_{\text{pos}} = \frac{1}{\sqrt{1/10^2 + 1/20^2 + 1/30^2}} = 8.57$$

If $\sigma_p \rightarrow \infty$ then $\sigma_{\text{pos}} \rightarrow \sigma_L$ (prior is very uninformative)

- Effect of uniform prior ?

- Update to the prior

$$\hat{\theta}_{\text{MAP}} = \theta_p + \frac{\sigma_p^2}{\sigma_p^2 + \sigma_L^2} \times (\hat{\theta}_{\text{MLE}} - \theta_p)$$

- If $\hat{\theta}_{\text{MLE}} = \theta_p$ then $\hat{\theta}_{\text{MAP}}$ is unchanged from the prior (but its variance is decreased)
- If σ_L is large compared to σ_p then $\hat{\theta}_{\text{MAP}} \approx \theta_p$ (noisy sensor)
- If σ_p is large compared to σ_L then $\hat{\theta}_{\text{MAP}} \approx \hat{\theta}_{\text{MLE}}$ (weak prior)

MAP Application 1: Berkeley traffic tracking system

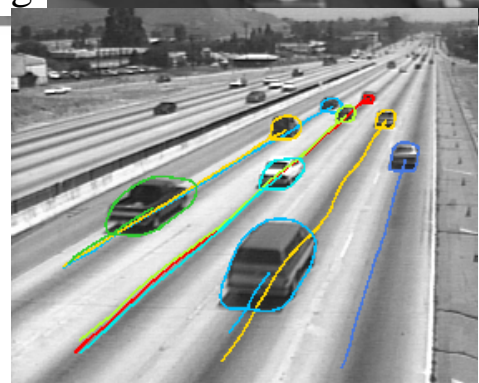
Tracks cars by:

- computing corner features on each video frame
- tracking corners (2D) from frame to frame

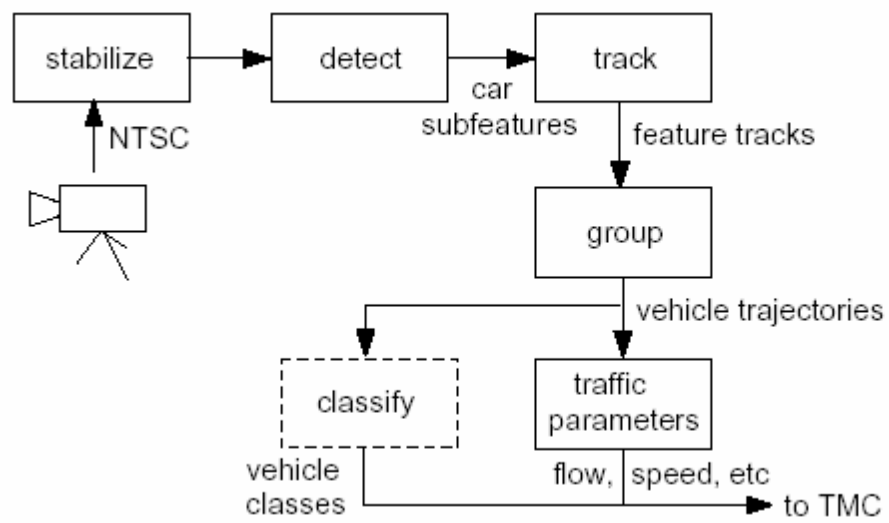


Objective: estimate image corner position $x(t)$

Tracking corner features

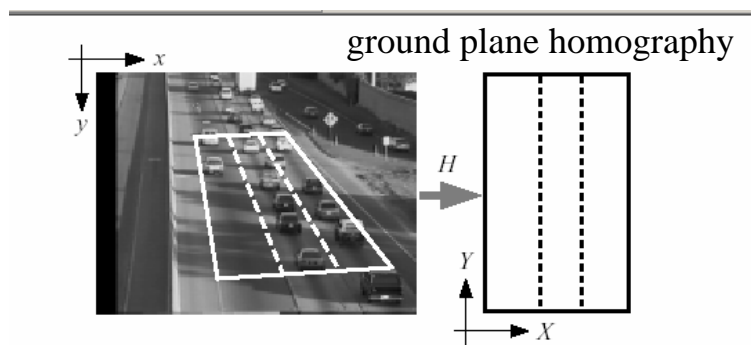
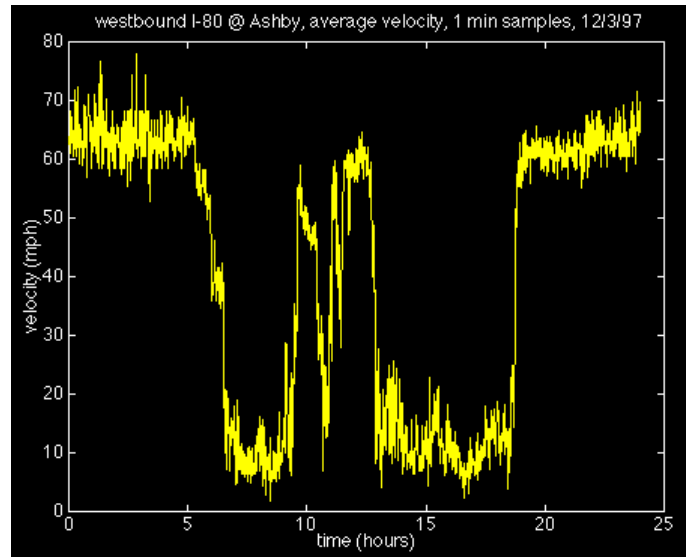
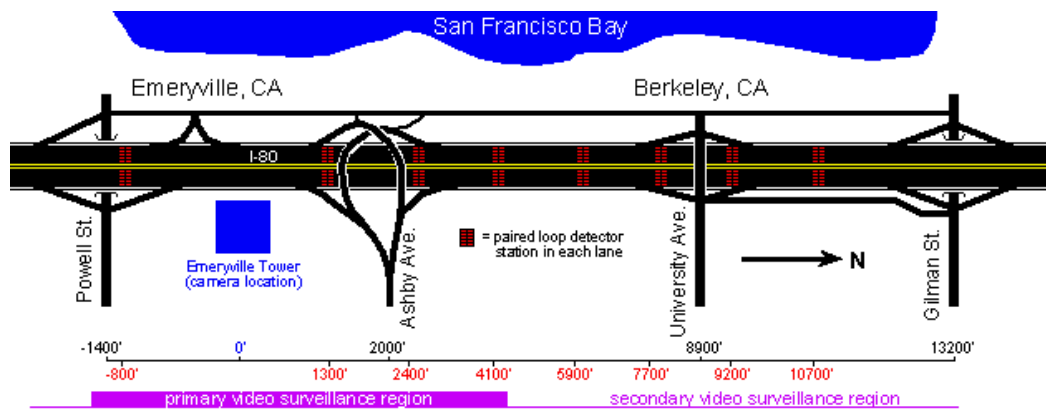


Berkeley visual traffic monitoring system

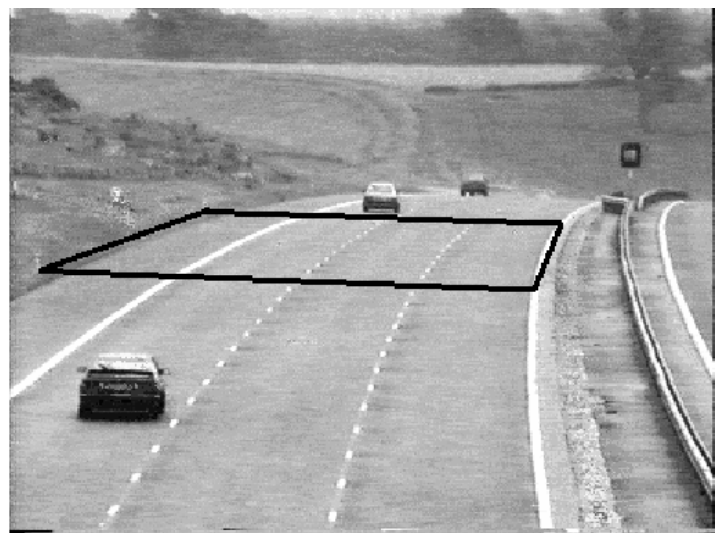


Traffic by day and by night



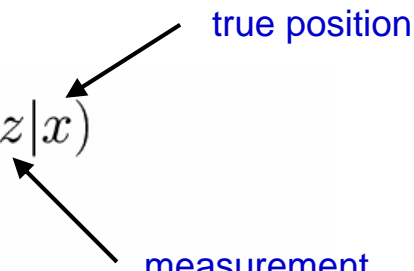


Velocity measurement gate



Tracking Corners

Measurement error model: $p(z|x)$



Prior information: $p(x)$

Inference (*posterior density*): $p(x|z)$

Bayes rule:

$$p(x|z) \propto p(z|x)p(x).$$

Corners need 2D Gaussians

Corner feature in 2D coordinates is a vector $\mathbf{x} = (x, y)^\top$.
Suppose $x \sim N(0, \sigma_x^2)$, $y \sim N(0, \sigma_y^2)$ then

$$p(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp \left\{ -\frac{1}{2} \left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} \right) \right\}$$

and the covariance matrix is

$$\Sigma = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$$

Corner fusion in 2D

Given a vehicle at position \mathbf{x} with prior distribution

$$N(\mathbf{x}_0, P_0)$$

and a corner measurement

$$\mathbf{z} \sim N(\mathbf{x}, R)$$

Summing information:

$$S = P^{-1} = R^{-1} + P_0^{-1}$$

Information weighted mean:

$$\hat{\mathbf{x}} = P \left(R^{-1} \mathbf{z} + P_0^{-1} \mathbf{x}_0 \right)$$

Corner fusion in 2D

Prior:

$$P_0 = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$$

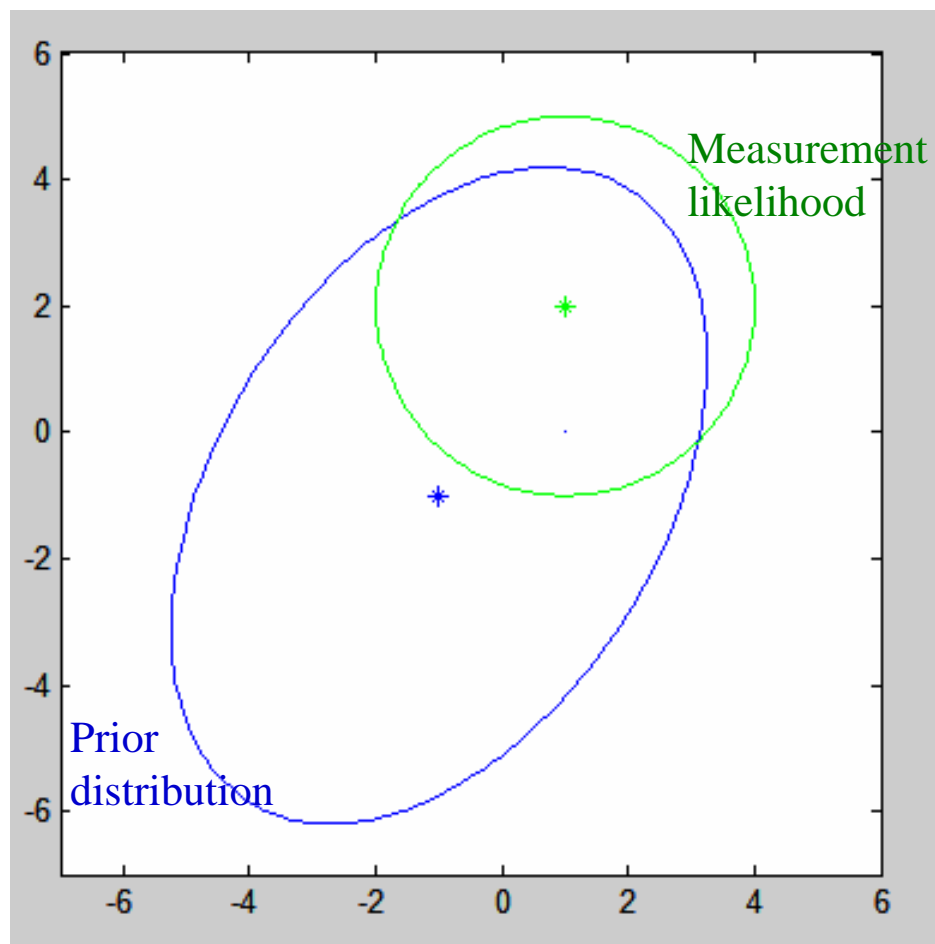
$$\mathbf{x}_0 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

Likelihood:

$$R = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{z} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

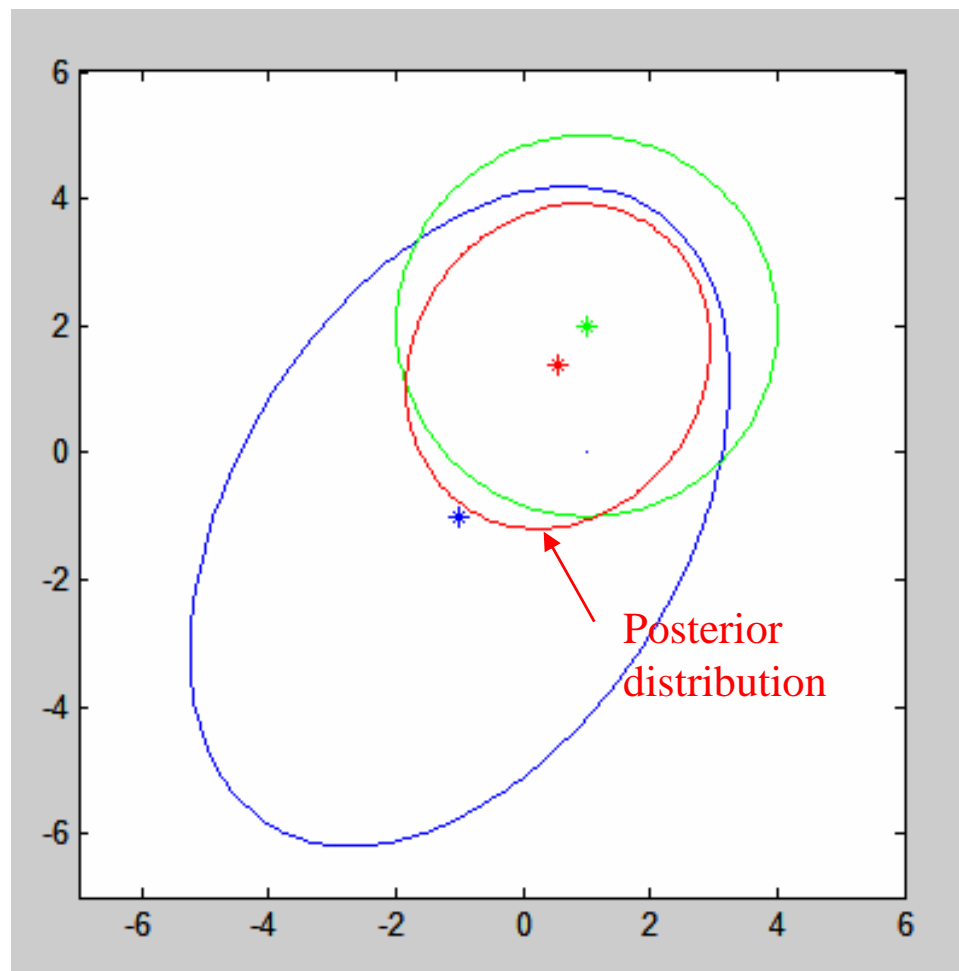
Isotropic
measurement error



Posterior:

$$P = \begin{pmatrix} 0.64 & 0.09 \\ 0.09 & 0.73 \end{pmatrix}$$

$$\mathbf{x} = \begin{pmatrix} 0.55 \\ 1.37 \end{pmatrix}$$



Where does the prior come from?

Use the posterior from the previous time step, “updated” to take account of the dynamics

$$\mathbf{x}_t - \mathbf{x}_{t-1} \sim \mathcal{N} \left(\begin{pmatrix} 0 \\ \mathbf{v} \end{pmatrix}, \sigma^2 \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right)$$

Iterate:

- prediction (posterior + dynamics) to give prior
- make measurement z
- fuse measurement and prior to give new posterior

Compute estimate and covariance for tracked corner

Prior:

$$P_0 = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}$$

$$\hat{\mathbf{x}}_0 = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

Dynamics:

$$P_D = \begin{pmatrix} 0.3 & 0 \\ 0 & 0.3 \end{pmatrix}$$

$$\mathbf{v} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Likelihood:

$$R = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\mathbf{z}_1 = \begin{pmatrix} 6 \\ 2 \end{pmatrix}$$

a) Prediction
at t=1

$$\tilde{P}_1 = P_0 + P_D = \begin{pmatrix} 2.3 & 1.0 \\ 1.0 & 3.3 \end{pmatrix}; \quad \tilde{\mathbf{x}}_1 = \hat{\mathbf{x}}_0 + \mathbf{v} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}$$

b) Fuse measurement
at t=1

$$P_1 = (\tilde{P}_1^{-1} + R^{-1})^{-1} = \begin{pmatrix} 0.674 & 0.076 \\ 0.076 & 0.750 \end{pmatrix}$$

$$\hat{\mathbf{x}}_1 = P_1 (\tilde{P}_1^{-1} \tilde{\mathbf{x}}_1 + R^{-1} \mathbf{z}_1) = \begin{pmatrix} 3.95 \\ 1.78 \end{pmatrix}$$

Berkeley tracker

In action ...



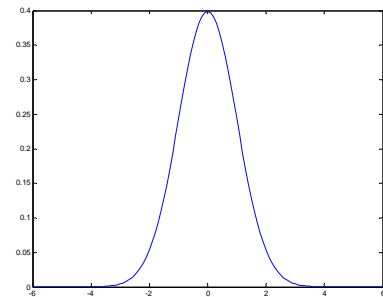
Confidence regions

- To quantify confidence and uncertainty define a confidence region R about a point x (e.g. the mode) such that at a confidence level $c \leq 1$

$$p(x \in R) = c$$

- we can then say (for example) there is a 99% probability that the true value is in R
- e.g. for a univariate normal distribution $N(\mu, \sigma^2)$

$$\begin{aligned} p(|x - \mu| < \sigma) &\approx 0.67 \\ p(|x - \mu| < 2\sigma) &\approx 0.95 \\ p(|x - \mu| < 3\sigma) &\approx 0.997 \end{aligned}$$



- In 2D confidence regions are usually chosen as ellipses

For an n dimensional normal distribution with mean zero and covariance Σ , then

$$\mathbf{x}^T \Sigma^{-1} \mathbf{x}$$

is distributed as a χ^2 -distribution on n degrees of freedom.

In 2D the region enclosed by the ellipse

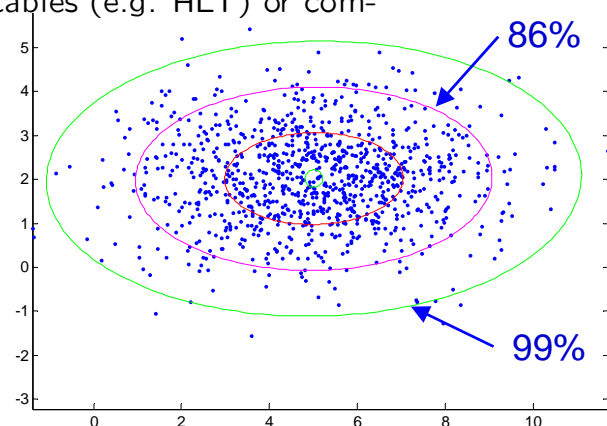
$$\mathbf{x}^T \Sigma^{-1} \mathbf{x} = d^2$$

defines a confidence region

$$p(\chi^2_2 < d^2) = c$$

where c can be obtained from standard χ^2 tables (e.g. HLT) or computed numerically.

| d | confidence |
|---|------------|
| 1 | 39% |
| 2 | 86% |
| 3 | 99% |



MAP Application 2: Image Restoration

Suppose sensor model is

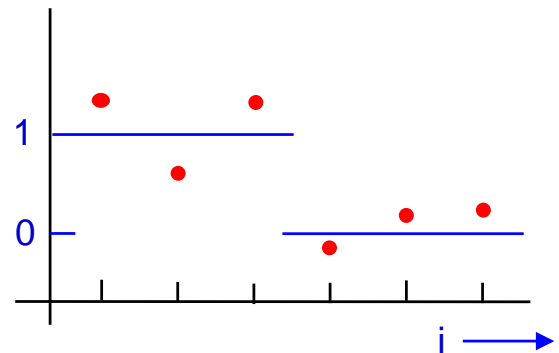
$$z_i = u_i + w_i$$

where

$$u_i \in \{0, 1\} \quad w_i \sim N(0, \sigma^2)$$

Then the likelihood is

$$p(\mathbf{z}|\mathbf{u}) \sim \prod_i e^{-(z_i - u_i)^2 / (2\sigma^2)}$$



Assume the prior has the form

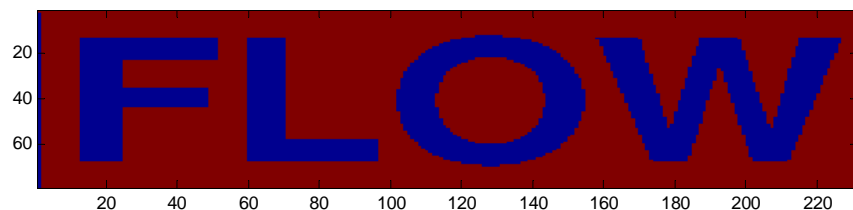
$$p(\mathbf{u}) \sim \prod_i e^{-\lambda(u_i - u_{i-1})^2}$$

then the negative log of the posterior $p(\mathbf{u}|\mathbf{z})$ is

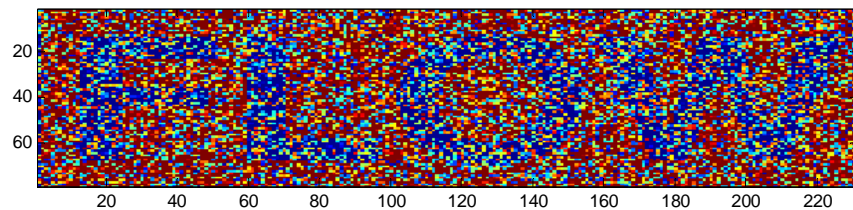
$$C = \underbrace{\sum_i (z_i - u_i)^2}_{\text{likelihood}} + \underbrace{d(u_i - u_{i-1})^2}_{\text{prior}} \quad \hat{\mathbf{u}}_{\text{MAP}} = \arg \min_{\mathbf{u}} C(\mathbf{u})$$

$d = 10$

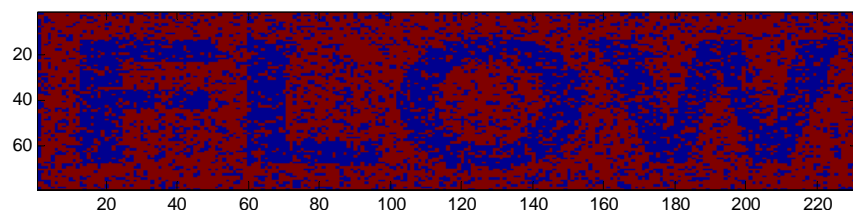
original



original
plus noise

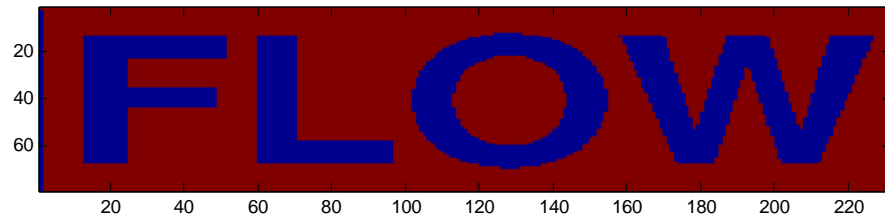


MAP
restoration

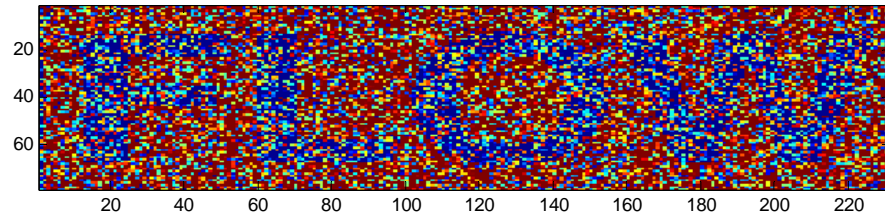


d = 60

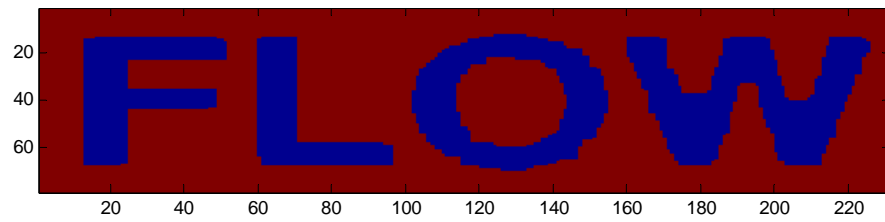
original



original
plus noise



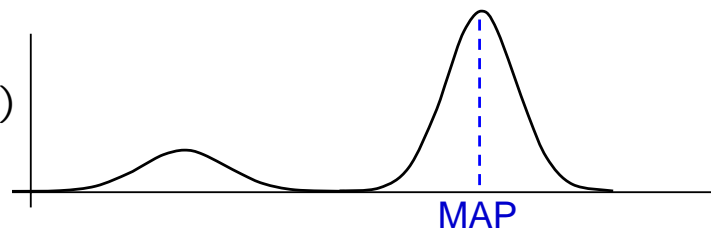
MAP
restoration



Loss Functions

Posterior

$$p(\theta|z) \sim p(z|\theta)p(\theta)$$

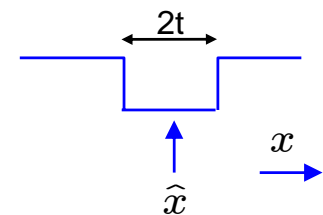


Example: hitting a target

$$\text{Loss function } L(\hat{x}, x) = \begin{cases} 1 & \text{if } |x - \hat{x}| > t \\ 0 & \text{if } |x - \hat{x}| \leq t \end{cases}$$

chosen
value

actual value



$$\text{Expected Loss } \hat{L}(\hat{x}) = \int p(x|z) L(x, \hat{x}) dx$$

Choose \hat{x} to minimize expected loss $\hat{x} = \arg \min_{\hat{x}} \int p(x|z) L(x, \hat{x}) dx$

Example 1: squared loss function

$$L(x, \hat{x}) = (x - \hat{x})^2$$

this results in choosing the mean value

Sketch proof

$$\mu = \arg \min_u \int (x - u)^2 p(x) dx$$

Differentiate w.r.t. u

$$\int 2(u - x)p(x) dx = 0$$

Choose \hat{x} to minimize expected loss $\hat{x} = \arg \min_{\hat{x}} \int p(x|z) L(x, \hat{x}) dx$

Example 2: absolute loss function

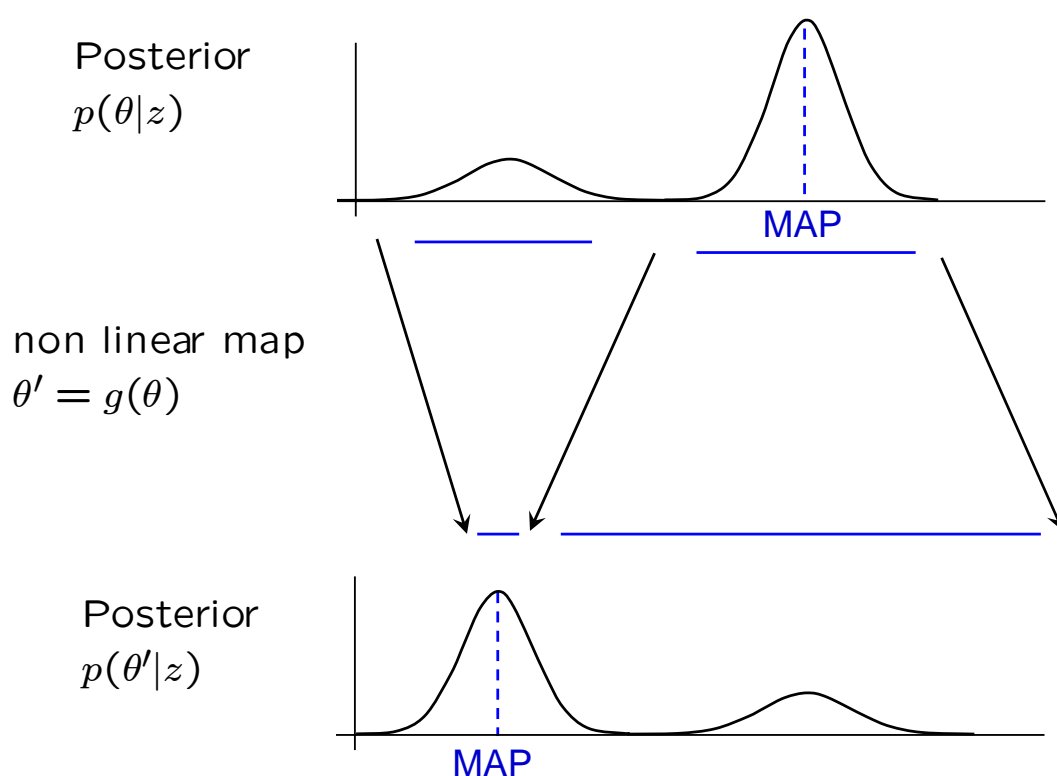
$$L(x, \hat{x}) = |x - \hat{x}|$$

this results in choosing the median value

$$x_{\text{med}} = \arg \min_u \int |x - u| p(x) dx$$

[proof: exercise]

Warning: MAP and change of parameterization



Problem:

- MAP finds maximum value of the posterior, but the posterior is a probability **density**

Solution:

- Loss functions correctly transform under change of coordinate (reparameterization)

Example of non-linear map: perspective projection

